

統合可能な作業を含む調理スケジュールリング問題に対する発見的解法

Heuristics for Cooking Scheduling Problems Involving Integratable Tasks

情報工学専攻 平野 敬祐
Information and System Engineering HIRANO Keisuke

あらまし： 飲食チェーン店での調理工程は簡単なもので構成され、誰でも調理が可能となっている一方で、単に注文の順に調理すると効率が悪く、調理手順の決定は、経験がなければ非常に難易度の高いものとなっている。本研究では、加熱時間に幅がある作業や、同時に処理することによって効率的になるなどの、現実で考慮したい制約などをモデル化する方法と、モデル化した問題に対する発見的解法を提案する。また、飲食チェーン店を想定し、オンラインでの実行を考え、スケジュールの提示と局所探索法を平行して行う。

キーワード： スケジュールリング、最適化

1. 序論

飲食店での調理者への作業の割り当てや作業順序は、料理の提供時間に直結する重要な問題である。作業順序によって、料理を早く提供することや同じ注文の料理を同時に提供することで、顧客の満足度を高めることができる。一方で、逐次的に入力される注文に対して調理しながら作業順序を決定するのは難しく、経験が必要であり、時間もかかる。

本研究では、資源制約付きスケジュールリング問題の中でも、ジョブを調理の作業、資源を調理器具（フライヤー、電子レンジなど）や調理者とする調理スケジュールリング問題を扱う。文献[1]では飲食店での顧客の満足度を向上させることを目的として、貪欲法のアルゴリズムによって調理スケジュールリング問題を解いている。本研究で想定する環境はファミリーレストランなどの飲食店であり、注文が入るまでどの料理を作るか与えられておらず、複数の注文が異なる時刻に与えられるため、オンラインスケジュールリングを行う。

2. 問題設定

2章では問題定義を行い、調理スケジュールリング問題独特のモデルとして、連続作業、統合作業について

説明する。また、調理者の作業場所間の移動を表現する方法も説明する。

2.1 制約

- 先行制約：ある料理において作業Aの処理時間を t_A 、作業A,Bの開始時刻を s_A, s_B とする。作業Bは作業Aの完了後に開始可能なとき、 $s_A + t_A \leq s_B$ である。各料理において、先行制約は作業間の半順序として与えられる。
- 資源制約：設備は同時に一つの作業しか処理することができない。同種の設備が複数ある場合、それぞれ別の設備として扱う。作業は複数の設備を同時に使用することがあり、例えば「焼く」という作業は、「調理者」と「IH」という二つの資源を使用する。
- 中断禁止制約：各作業を中断し、他の作業の終了後に続きを行うことは認めない。

2.2 連続作業

調理を行うにあたり、一つの作業の途中で必要な資源が変化する作業がある。また、ある作業の完了後すぐに次の作業を開始しなくてはいけないこともある。これらをモデル化する方法を説明する。

例として揚げ物を考える。揚げ物は表1のように作業を細かく分けることができる。揚げ物は食材をフライヤーに入れてから一定時間加熱後に取り出す作業が必要となる。加熱時間には幅を持たせることができる。このような処理のモデル化を考える。

表 1: 揚げ物の例

作業	処理時間	使う資源
フライヤーに入れる (A)	5秒	調理者, フライヤー
揚げる (B)	100~120秒	フライヤー
フライヤーから取り出す (C)	5秒	調理者, フライヤー

表1の例では、「揚げる(B)」は「フライヤーに入れる(A)」の完了後すぐに開始されるようにそれぞれの開始時刻を制限する必要がある。そこでBはAの開始時刻から5秒後以降に開始し、AはBの開始時刻から5秒前以降に開始するとする。このように設定することで、作業Aの完了後すぐに次の作業Bを開始する。

また、「フライヤーから取り出す(C)」はBの開始時刻から100秒後以降に開始し、BはCの開始時刻から120秒前以降に開始する。このように設定することによって、CはBの開始時刻から100秒から120秒後に開始することになり、間接的にBの作業時間に幅を持たせることができる。この設定は、ホイススケジュールリング問題に対する研究[2]でも用いられている。

以上の設定をすることで、細分化された作業を連続的に、処理時間に幅を持たせて行うように制約を与えることができる。このような、ある作業が終了したのちに一定時間以内に次の作業を開始する制約を連続制約、それらの作業を連続作業と呼ぶ。

なお、連続作業が同じ設備種の設備を使用する場合、それらの作業は同じ設備を使用しなくてはならない。

2.2 統合作業

調理スケジュールリングでは、内容が同じ作業を同時に行うと効率が良い場合がある。例えば「トングを用いて盛り付けをする」作業の場合、「トングを持つ」、「盛り付ける」、「トングを置く」という三つの小作業となる。この一連の作業を二度行うことを考えたとき、単純に二度繰り返すこともできるが、「トングを持つ」、「トングを置く」という作業を二度ずつ行うのは効率が悪い。そこで、これらの作業を一度に行うことを考えると、図1のように別々に行うより同時に行った方が効率が良いことがわかる。このように同じ内容の作業をひとまとめに行うことを、作業の統合と呼び、統

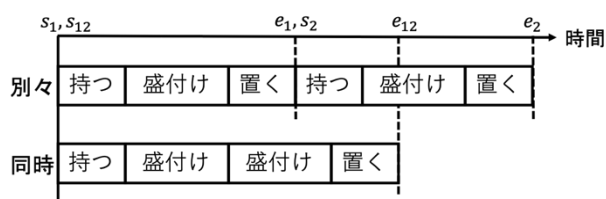


図1: 内容が同じ作業を別々に行う場合と同時に行う場合の作業時間比較

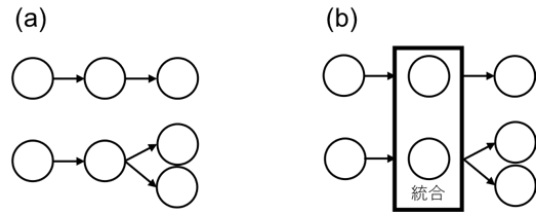


図2: 作業の統合を行なったときの制約の変化

合された作業を統合作業と呼ぶ。

作業は負荷を表す重みを設定し、作業の処理時間は、図1の「トングを持つ」、「トングを置く」時間のような固定でかかる時間と、「盛り付ける」時間のような負荷（作業量）によって変動する処理時間を組み合わせて表現する。作業を統合すると作業の開始時刻が同じになり、その作業の負荷重みをそれら作業の負荷重みの和に更新することで、処理時間を正しく計算できる。作業を統合すると作業全体での処理時間は短くなり、より早く処理することができる。一方で、作業を統合するデメリットとして、元々制約が図2aのように与えられているものが、図2bのように統合された作業にかかる制約が相互に与えられるため、ある料理の完成時刻が早くなる一方で別の料理の完成時刻が遅くなる可能性もあることに注意する必要がある。

2.3 作業場所移動の表現

調理器具や作業場所はそれぞれ離れた場所にあり、調理者が続けて処理する作業の作業場所が異なるとき移動を行う。この移動を表現するために段取り替え時間を導入する。段取り替え時間は、連続して処理される作業の組み合わせに応じて発生する追加の処理時間であり、本研究では、調理者が連続して処理する作業の組み合わせに応じて、後に処理する作業の処理時間に、調理者の移動時間として段取り替え時間を加える。各作業内容に作業ポジション情報を与え、このポジション間の移動距離を距離行列として与える。

例として、図3のように、ある調理者が作業A,B,Cの順に処理するとする。作業Bと作業Cのポジションは同じであり、作業Aのみ異なるとする。このとき、作業Aは前に処理する作業が存在しないので移動を考慮しなくて良く、段取り替え時間は発生しない。作業Bは、作業Aのポジションaから、作業Bのポジションbへの移動が発生するので、処理時間にポジションaからbへ

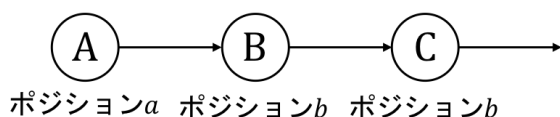


図 3: 異なるポジションの作業を連続で処理する例

の移動時間を表す段取り替え時間を加える。作業Cは、直前の作業Bと同じポジションでの作業なので、移動が発生せず、段取り替え時間は発生しない。

3. 解の評価手法

本研究では各伝票内の最後に提供された料理の提供にかかった時間の総和の最小化を目的とする。この目的関数を計算するためには、各作業の開始時刻を決定する必要がある。しかし、本研究で扱う問題は連続制約や作業の統合によって作業の開始時刻を決定することが困難である。そこで、作業を頂点、各作業間の開始時刻の制約を辺として表す有向グラフを用いて各作業の開始時刻を決定する。

作業Bは作業Aの開始時刻より t_A 秒後以降に開始できるとき、頂点Aから頂点Bに長さ t_A の有向辺を張る。これによって先行制約や資源制約が表現でき、辺の正負を反転させることで連続制約を表現できる。作業の統合は開始時刻を同一に揃えるので、統合される作業間に長さ0の辺を双方向に張ることで表現する。なお、作業の統合や設備の処理順序によって作業の処理時間や段取り替え時間が変動するので、都度辺の長さを更新する必要がある。

このグラフに開始時刻0、処理時間0を表すスーパーノードを加え、スーパーノードから各料理の先頭ノードに対し最早開始時刻長の辺を張り、スーパーノードからの最長路を計算すると、各ノードの最長路長が作業の開始時刻となる。

標準的なスケジューリング問題では、グラフは非巡回の有向グラフとなり、最長路を求めるのは簡単である。しかし、連続制約が存在するため長さが負の辺が存在し、かつ負閉路が存在する。一方で閉路長が正の閉路は存在しないので、各辺の長さの正負を反転させ、ベルマンフォード法を用いて最短路を求めることで、元のグラフの最長路が求められる。以降の説明では辺の長さを正負反転したグラフを用いる。

4. 局所探索法

各設備が処理する作業の順序を決定することで、3章で説明した方法で各作業の開始時刻を決定でき、スケジュールを評価できる。まず、入力された注文から資源制約や作業の統合以外の制約に関する辺を張る、解の構築や局所探索では、処理順序の決定や変更のたびに辺の張り替えを行い、最短路計算によってスケジュールを決定し、解を評価する。4章では解表現と解の構築、近傍操作について説明する。

4.1 解表現

各設備の処理順序によってスケジュールが決定するため、設備ごとの処理順序を解としたい。しかし、本研究では作業が複数の設備を使用することがあるため、不都合が生じる。例えば、ある設備では作業A,Bの順に処理し、別の資源では作業B,Aの順に処理するという順序を与えることも可能であるが、そのような処理順序は明らかに実行不可能である。

そこで、同じ設備を使用する作業間ではその設備の処理順序が保たれるような作業の順序集合のリストを解表現として用いる。このリストに各作業の使用設備を組み合わせることで各設備の処理順序が決定される。このリストの並びを変える操作を近傍操作として局所探索法を行うことで、前述のような実行不可能な解の評価を避けることができる。なお、解のリストは常に開始時刻順に保つ。

4.2 解の構築

解の構築は、5章で述べる各作業の納期と最遅開始時刻の差を優先度として、優先度の降順に解を表すリストに対して前から順に挿入を試し、実行可能なものが見つかった時点で次の作業を試す。

4.3 近傍操作

局所探索では単に解のリスト上での作業の位置を別の位置に変える挿入近傍と作業を別の統合可能な作業に統合することを試す統合近傍の二つを用いる。ただし、先行制約に違反する位置への作業の挿入や統合は、グラフ上で必ず負閉路が生じ、実行不可能となるので、そのような解は評価せず、近傍内の改善解に移動する。また、一度統合された作業は統合解除することはあまり無いと考え、すでに統合されている作業を動かすような近傍操作は行わない。連続作業に関し



図 4: 探索の流れ

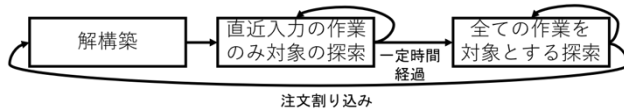


図 5: 全体の流れ

ては挿入近傍では一括して動かすことを考える。

探索は、図 4 の流れで行い、挿入近傍ではクリティカルパスを用いて探索の領域を制限する。また、全体の流れは図 5 の流れで、オンラインスケジューリングによって逐次的に最適化されていることから、注文発生時は新しい注文の作業に絞って探索する。

解の評価時間はグラフのサイズに依存するため、以下の探索をオンラインで行う。暫定解のうち、開始時刻が現在時刻から一定時間以内の作業は、開始時刻を固定し、スケジュールの提示とグラフからの除去を行う。まだ固定していない作業のみに対して近傍操作を行うことで解評価を高速に行う。

5. 数値実験

オンラインでの実験を行った。注文数9の実験では、料理が18種、作業内容が96種、各注文の作業数が26~183、各注文の先行制約、連続制約数が35~282、設備種が9、設備数が20の入力を用い、注文が22~123秒の間隔で入力される。各注文の納期は入力から720秒後とし、納期違反すると、違反度の二乗のペナルティがかかる。10秒ごとに探索を中断し、その時点から10秒以内に開始する作業を提示し、開始時刻が変動しないように固定し、実験を行なった。

実験を行ったところ、制約を満たし、一部の作業は統合されたスケジュールが得られた。図 6 は局所探索による評価値とペナルティのグラフで、図 6 上は注文あたりの評価値の平均を表し、縦軸は対数軸となっている。注文が入力され、解を構築した時点では評価値が一度悪化している。その後の探索で改善しているが、注文が増えるにつれ、解を改善することができず、注文が全て入力された後に改善されている。これは、作

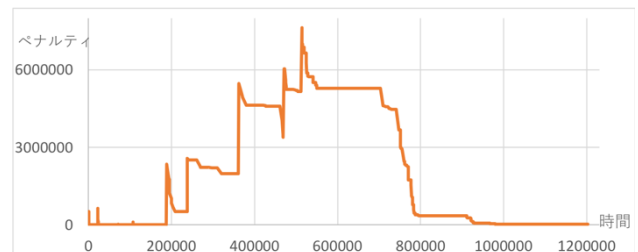
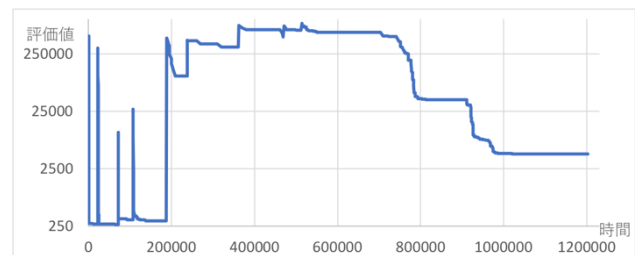


図 6: 評価値 (上) とペナルティ (下) のグラフ

業が増え十分に探索できていないと考えられる。一方で、探索の後半では、4 章の解評価の効率化によってグラフのノード数が減り、探索が進むようになったことで改善されたと考えられるが、改善できない時間が多く、納期違反したスケジュールが得られた。

6. 結論

本研究では、連続制約や作業の統合などを考慮した調理スケジューリング問題をモデル化し、局所探索法を用いたオンラインスケジューリングアルゴリズムを提案した。オンラインの性質を活かした解評価の効率化によって、各時点での作業数が 300 程度に抑えられれば、探索によって良いスケジュールを得ることができている一方で、それを超える料理、注文の入力に対しては探索しきれず、あまり良いスケジュールは得られなかった。評価の高速化や、新たな近傍の設計、改善が今後の課題として考えられる。

参考文献

- [1] Yuki KIMURA et al., An approach to cooking process scheduling for a family restaurant, Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol. 12, No. 3, pp. 1-12 (2018)
- [2] 野々部 宏司, ホイストスケジューリング問題に対する発見的手法, スケジューリング・シンポジウム 2018 講演論文集, pp. 139-143 (2018)